

REMARKS/ARGUMENTS

Claims 1-20 are pending in the present application. In this amendment, Applicants have amended claims 7, 13, and 16. Applicants are not conceding that the subject matter encompassed by claims 7, 13, and 16, prior to this Amendment, is not patentable over the art cited by the Examiner. Claims 7, 13, and 16 were amended in this Amendment solely to facilitate expeditious prosecution of the application. Applicants respectfully reserve the right to pursue claims, including the subject matter encompassed by claims 7, 13, and 16 as presented prior to this Amendment and additional claims in one or more continuing applications. Reconsideration of the claims is respectfully requested.

I. New Ground of Rejection Under 35 U.S.C. § 101

The Examiner has rejected claims 13-18 under 35 U.S.C. § 101 as being directed towards non-statutory subject matter. This rejection is respectfully traversed.

The Examiner states:

In the current application, claims 13-18 recite method steps for executing a workflow. While the preamble recites computer terminals and a network, mere recitation in the preamble is not enough to tie one of the steps of the process to another statutory class. Thus, claims 13-18 fail the first prong of the new Federal Circuit decision since it is not tied to another statutory class and can be performed without the use of a particular apparatus and it is respectfully submitted that claims 1-5 are directed to non-statutory subject matter.

Examiner's Answer, December 9, 2008, pages 5-6.

Applicants have amended claims 13 and 16 to incorporate a workflow server, which is hardware. Therefore, since the claims are tied to another statutory class, the claims are directed towards statutory subject matter.

The Examiner rejected claims 7-10 and 12-15 under 35 U.S.C. § 102(e) as being anticipated by *DeFrancesco, Jr.* et al (US Patent No. 6,505,176, filed January 7, 2003, hereafter referred to as "*DeFrancesco*"). This rejection is respectfully traversed.

DeFrancesco does not anticipate Applicants' claims because *DeFrancesco* does not teach executing skipped steps. Once a step is skipped in *DeFrancesco*, that step is not executed at a later time.

Applicants' claim 7 describes the workflow including nodes that may be skipped. Recovery nodes indicate points in the workflow where skipped nodes are executed.

Applicants' claim 10 describes means for establishing a node in the workflow that may be skipped. There are points on the workflow where transactions that were part of the skipped node are executed.

Applicants' claim 13 describes designating at least one node in the workflow that may be skipped during the execution of the workflow. A recovery node is designated that indicates a re-execution point in the workflow where the persons in charge of a previously skipped node re-execute the business processes of that node.

Applicants' claims describe nodes that may be skipped. The claims describe points in the workflow where nodes that were originally skipped are executed.

DeFrancesco teaches one environment in which the invention of *DeFrancesco* can be used. The environment is within a lending institution for processing loan and credit applications. Steps used to process loan applications, and the order in which these steps are processed, vary widely among lending institutions. See *DeFrancesco*, column 2, lines 4-12.

A workflow is defined that includes these steps. A user can build a workflow definition 404 by defining and selecting workflow steps. *DeFrancesco*, column 8, lines 12-16. There are three types of steps. The steps are either "normal", "exception", or "automatic" steps. *DeFrancesco*, column 8, lines 22-25. The workflow management system evaluates the steps to determine their status (incomplete, not applicable (N/A), complete, or waived), and determines which steps are next activated. The steps that have not already been skipped or waived have a status of either "incomplete" or "complete". The steps that are skipped are tagged with a N/A status.

"Skip tests are used to determine the presence of a specified criteria that would cause the associated step 418 to be tagged with a status of N/A as shown by status block 416. Steps 418 having a status 416 of N/A do not apply to the workflow, and are therefore skipped. For example, one step may be to send out a decline letter to the applicant. However, this step should be skipped if the applicant is granted a loan." *DeFrancesco*, column 9, lines 57-64. Thus, as the example makes clear, skipped steps are not repeated because they are not necessary. It is not necessary, and would make no sense, to send a decline letter to an applicant that has already been granted a loan. There would be no reason to attempt to complete an unnecessary step.

DeFrancesco provides additional information about skip tests in column 10, lines 23-25, which states: “Skip Tests - Tests that are tested to determine if there is a special circumstance that would cause the step to not apply to the workflow and therefore be skipped.” If a test does not apply, it should not be executed at a later time.

The completion tests are used to determine if a step has been completed. *DeFrancesco*, column 10, lines 20-25. Once the step has been completed, its status is changed from “incomplete” to “complete”. The completion tests are used for steps that have not been skipped. Skipped steps do not have a status of either “complete” or “incomplete”. If a step has either a status of “complete” or “incomplete”, that step was not skipped.

“Exception steps are used to manage any exceptions encountered in the normal processing of credit applications. Exception steps are typically configured to follow the actual step that causes the exception.” *DeFrancesco*, column 8, lines 32-35. “A exception test set is used to determine if an exception should be made.” *DeFrancesco*, column 11, lines 9-10.

Exception tests are tests that are only applicable for exception-type steps and act as both skip and completion tests for exception steps. See *DeFrancesco*, column 10, lines 22-23. When determining whether a “normal” or “exception” step should be skipped, *DeFrancesco* uses the skip tests. “If the process step is of the type exception 414, the completion and skip tests 422 are the same. Accordingly, the rules 412 associated with an exception step are both the skip and completion rules.” *DeFrancesco*, column 13, lines 38-41. Therefore, when determining whether an exception step should be skipped, the exception tests act just like the skip tests described above. Therefore, the discussion above regarding the skip tests applies to exception tests as well.

Thus, according to *DeFrancesco*, an exception test can be used to determine whether a step should be skipped or whether it should be completed. *DeFrancesco* does not describe executing a skipped step until that step is completed. Throughout *DeFrancesco*, skipped steps are never described as being executed. Skipped steps are described as being skipped and marked “not-applicable”.

Steps that need to be executed are marked as being either complete or incomplete. When a step has been executed, it is marked as being complete. When a step that needs to be executed has failed, it is marked as being incomplete. Attempts are then made to execute these incomplete steps until they are executed. Once executed, their status is changed from incomplete to complete.

The status of a step provides an additional indication of whether the step was skipped. Skipped steps, whether normal steps or exception steps, never have a status of “incomplete” or “complete”. Skipped steps are always described as having the status “not-applicable”, or “N/A”.

Exception steps can either be skipped, or they need to be completed. If they are skipped, they have a “not-applicable” status, and are not executed. If the step needs to be completed, the completion tests are used to determine if the step has been completed. The exception steps that need to be completed, i.e. the steps that have not been skipped, have an “incomplete” status until they are completed; when they are completed they have a “complete” status.

The Examiner asserts that *DeFrancesco* teaches executing skipped steps. Specifically, the Examiner states that “Where there is an exception, both the skip and complete tasks apply and at the failure of either test, a user is prompted to re-execute the failed processes.”

Examiner’s Answer, page 7. Applicants respectfully disagree that this is what is taught by *DeFrancesco*.

DeFrancesco, column 13, lines 38-50, states:

If the process step is of the type exception 414, the completion and skip tests 422 are the same. Accordingly, the rules 412 associated with an exception step are both the skip and completion rules. Thus, for exception steps, if at least one of the rules fail, an exception is indicated and the status 416 for the step is incomplete. This will prompt attention from a user, that action is required to complete the process step. Once the user performs the required action, these tests will be executed again. If at that time, all of the tests pass, the step will be tagged with a complete status. If all of the tests for an exception step pass the first time through, there is no exception and the rule is skipped. The status 416 for a skipped exception rule is non-applicable (N/A).

The section of *DeFrancesco* quoted above first teaches that the skip tests and completion tests are the same for the exception-type steps. This section then goes on to describe what happens if a test (described here as a “rule”) fails for an exception-type step. The tests that might fail are the completion tests. If an exception-type test fails, the step is marked as being “incomplete”. Steps that are marked as being incomplete prompt attention from a user. Once the user performs the action, the completion tests are executed again until all completion tests pass.

Furthermore, it does not make sense that a completion test would be performed on a step that has already been skipped. If a step has been skipped, that step is not-applicable, and is not performed. A completion test would be used only on steps that need to be performed.

The section of *DeFrancesco* quoted above does not describe executing skipped steps until the skipped steps have been completed. Skipped steps never have a status of “complete”. They always have a status of “N/A”. Because the step described above is tagged with a “complete” status, the step was not skipped.

DeFrancesco does not anticipate Applicants’ claim 7 because *DeFrancesco* does not teach the workflow including nodes that may be skipped and recovery nodes that indicate points in the workflow where skipped nodes are executed.

DeFrancesco does not anticipate Applicants’ claim 10 because *DeFrancesco* does not teach means for establishing a node in the workflow that may be skipped, where there are points on the workflow where transactions that were part of the skipped node are executed.

DeFrancesco does not anticipate Applicants’ claim 13 because *DeFrancesco* does not teach designating at least one node in the workflow that may be skipped during the execution of the workflow, where a recovery node is designated that indicates a re-execution point in the workflow where the persons in charge of a previously skipped node re-execute the business processes of that node.

Because *DeFrancesco* does not teach all of the features of Applicants’ claims, *DeFrancesco* does not anticipate Applicants’ claims. The remaining claims depend from the independent claims discussed above and are patentable for the reasons given above.

The Examiner rejected claims 1-6, 11, and 16-20 under 35 U.S.C. § 103(a) as being unpatentable over *DeFrancesco* in view of *Gabbita*, et al. (US Patent No. 6,349,238, filed February 19, 2002, hereafter referred to as “*Gabbita*”). This rejection is respectfully traversed.

The combination of *DeFrancesco* and *Gabbita* does not render Applicants’ claims obvious because the combination does not teach executing skipped steps or performing reassignment of the activities.

Claim 1 describes activities that may be skipped and re-execution points in the workflow where previously skipped activities are executed.

Claim 3 describes means for skipping part of the process flow by skipping one of the operators to whom a transaction is assigned, and means for reassigning the skipped transaction to the skipped operator wherein the skipped transaction is executed.

Claim 5 describes skipping the transaction assigned to a person and then assigning re-execution of the transaction to the skipped person. The re-execution is assigned at a predetermined timing in the workflow.

Claim 16 describes skipping the transaction assigned to a person and then assigning re-execution of the transaction to the skipped person. The re-execution is assigned at a predetermined timing in the workflow.

Claim 19 describes indicating that at least one of the nodes of the workflow may be skipped if the business processes of the node cannot be completed when the workflow defines that the node is to be executed. The recovery node is established where the persons in charge of skipped nodes re-execute the business processes.

Claim 20 describes skipping one or more activities assigned to persons, and assigning re-execution of the skipped activities to the persons whose assigned activities have been skipped.

The Examiner states that *DeFrancesco* teaches re-performing skipped activities. The Examiner goes on to state that *DeFrancesco* does not teach performing reassignment of the activities. The Examiner relies on *Gabbita* to teach the features believed missing from *DeFrancesco*.

The Examiner relies on *Gabbita*, column 3, lines 7-14, and column 29, lines 20-67, as teaching reassigning workflow processes. These sections of *Gabbita* teach a user being able to transfer and re-assign a workflow step. The user can use a remote workstation to do this.

Applicants' claim 1 describes re-execution points in the workflow where previously skipped activities are executed. These skipped activities are those that were skipped. As discussed above, *DeFrancesco* does not teach executing skipped activities. *Gabbita* also does not teach executing skipped activities. *Gabbita* teaches merely that a user can transfer a workflow step and assign it to another. Transferring a workflow step is not the same as executing activities that were skipped.

Neither *DeFrancesco* nor *Gabbita* teaches activities that may be skipped and re-execution points in the workflow where previously skipped activities are executed. Therefore, the combination of *DeFrancesco* and *Gabbita* does not render Applicants' claim 1 obvious.

The remaining claims describe features that are similar to the features of Applicants' claim 1 discussed above and are patentable for the reasons given above.

Because neither *DeFrancesco* nor *Gabbita* teaches the features of Applicants' claims, the combination of *DeFrancesco* and *Gabbita* does not render Applicants' claims obvious.

It is respectfully urged that the subject application is patentable over the cited references and is now in condition for allowance.

The Examiner is invited to call the undersigned at the below-listed telephone number if in the opinion of the Examiner such a telephone conference would expedite or aid the prosecution and examination of this application.

DATE: February 9, 2009

Respectfully submitted,

/Lisa L.B. Yociss/

Lisa L.B. Yociss
Reg. No. 36,975
Yee & Associates, P.C.
P.O. Box 802333
Dallas, TX 75380
(972) 385-8777
Attorney for Applicants